



**TUGAS AKHIR TT 090361**

# **RANCANG BANGUN SISTEM PENGKONDISIAN SUHU MENGGUNAKAN ELEMEN PELTIER DENGAN SISTEM MONITORING PADA SMARTPHONE ANDROID**

**ANDAN TUNJUNG PANGESTI  
NRP 2411.031.015**

**Dosen Pembimbing 1 :  
Dr. Ir. Purwadi Agus Darwito, M.Sc**

**Dosen Pembimbing 2 :  
Ir. Heri Joestiono, MT.**

**Program Studi DIII Metrologi dan Teknik Instrumentasi  
Jurusan Teknik Fisika  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2014**



FINAL PROJECT TT 090361

***BUILD AND DESIGN CONDITIONING SYSTEM  
OF TEMPERATURE USING PELTIER ELEMENT  
WITH MONITORING SYSTEM ON ANDROID  
SMARTPHONE***

ANDAN TUNJUNG PANGESTI  
NRP 2411.031.015

Advisor Lecturer 1 :  
Dr. Ir. Purwadi Agus Darwito, M.Sc

Advisor Lecturer 2 :  
Ir. Heri Joestiono, MT.

Diploma 3 of Metrology And Instrumentation Engineering  
Department of Engineering Physics  
Faculty of Industrial Technology  
Sepuluh Nopember Institute of Technology  
Surabaya 2014

**LEMBAR PENGESAHAN**  
**TUGAS AKHIR**

**RANCANG BANGUN SISTEM PENGKONDISIAN  
SUHU MENGGUNAKAN ELEMEN PELTIER  
DENGAN SISTEM MONITORING PADA  
SMARTPHONE ANDROID**

**OLEH:**  
**ANDAN TUNJUNG PANGESTI**  
**NRP. 2411.031.015**

**Surabaya, 17 Juli 2014**  
**Menyetujui,**

Dosen Pembimbing 1,



**Dr. Ir. Purwadi Agus Darwito, M.Sc.**  
**NIPN:19620822 198803 1 001**

Dosen Pembimbing 2,



**Ir. Heri Joestiono, MT.**  
**NIPN:19531116 198003 1 001**

**Mengetahui,**

Ketua Jurusan  
Teknik Fisika FTI ITS



**Dr. Ir. Totok Soehartanto, DEA**  
**NIPN:196503091990021001**

Kepala Program Studi  
D3 Metrologi dan Instrumentasi



**Dr. Ir. Purwadi Agus Darwito, M.Sc.**  
**NIPN:19620822 198803 1 001**

**LEMBAR PENGESAHAN**  
**TUGAS AKHIR**

**RANCANG BANGUN SISTEM PENGKONDISIAN  
SUHU MENGGUNAKAN ELEMEN PELTIER  
DENGAN SISTEM MONITORING PADA  
SMARTPHONE ANDROID**

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Ahli Madya  
pada

Program Studi D3 Metrologi dan Instrumentasi  
Jurusan Teknik Fisika  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember

Oleh :

**ANDAN TUNJUNG PANGESTI**  
**NRP. 2411.031.015**

Disetujui oleh Tim Penguji Tugas Akhir :

1. Dr. Ir. Purwadi Agus D. M.Sc. .... (Dosen pembimbing I)
2. Ir. Heri Joestiono, MT. .... (Dosen Pembimbing II)
3. Ir. Tutug Dhanardono, MT. .... (Dosen Penguji I)
4. Ir. Jerri Susatio, MT. .... (Dosen Penguji II)
5. Irwansyah, ST. MT. .... (Dosen penguji III)

**SURABAYA**  
**Juli 2014**

## **“RANCANG BANGUN SISTEM PENGKONDISIAN SUHU MENGUNAKAN ELEMEN PELTIER DENGAN SISTEM MONITORING PADA SMARTPHONE ANDROID”**

**Nama Mahasiswa** : Andan Tunjung Pangesti  
**NRP** : 2411 031 015  
**Program Studi** : D3 Metrologi dan Instrumentasi  
**Jurusan** : Teknik Fisika FTI-ITS  
**Dosen Pembimbing** : 1. Dr. Ir. Purwadi Agus Darwito, M.Sc.  
2. Ir. Heri Joestiono, MT.

### **Abstrak**

Berdasarkan analisa badan organisasi kesehatan dunia menunjukkan bahwa efek buruk asap rokok lebih besar bagi perokok pasif dibandingkan perokok aktif. Melihat keadaan tersebut, maka pada tugas akhir ini dirancang suatu alat untuk mengkondisikan ruangan yang terdapat asap rokok tersebut. Proses pengkondisian ruangan ini memanfaatkan *peltier* sebagai elemen pendingin. Arduino Mega ADK dan *smartphone* berbasis Android digunakan untuk sistem kontrol dan layar tampilan. Sistem komunikasi yang digunakan pada Android adalah USB. Hasil pengukuran ditampilkan pada *smartphone* dalam bentuk *logger*. Hasil pengujian efek *peltier* terhadap ruangan dipengaruhi oleh suhu luar ruangan. Sehingga besar suhu yang dihasilkan ditentukan oleh suhu luar ruangan. Berdasarkan data pengujian sensor, didapatkan rata-rata prosentasi *error* alat ukur suhu adalah sebesar 0.04%, rata-rata prosentasi presisi adalah sebesar 82.58% dan rata-rata akurasi yaitu 99.58% dengan *range* pengukuran 30°C sampai 57°C.

**Kata Kunci:** *Pengukuran Suhu, Sensor LM35, Elemen Peltier, Android, Logger*



# ***“BUILD AND DESIGN CONDITIONING SYSTEM OF TEMPERATURE USING PELTIER ELEMENT WITH MONITORING SYSTEM ON ANDROID SMARTPHONE ”***

***Name*** : Andan Tunjung Pangesti  
***NRP*** : 2411 031 015  
***Study Program*** : D3 Metrology and Instrumentation  
***Departement*** : Engineering Physics FTI-ITS  
***Advisor Lecturer*** : 1. Dr. Ir. Purwadi Agus Darwito MSc.  
2. Ir. Heri Joestiono, MT.

## ***Abstract***

*Based on World Health Organization body analysis showed that the adverse effects of cigarette smoke is greater for passive smokers compared to current smokers. Seeing the situation, then the final project is a tool designed to condition a room that contained the smoke. This space conditioning process utilizes a Peltier cooling element. Arduino Mega ADK and Android-based smartphones are used for control and display systems. Communication system used in Android is USB. The measurement results displayed on the smartphone in the form of a logger. The results of trying on the Peltier effect is influenced by the temperature of the room outdoors. So large that the temperature generated is determined by the temperature outdoors. Based on sensor test data, obtained an average percentage error of temperature measuring devices is 0.04%, the average percentage amounted to 82.58% precision and the average accuracy is 99.58% with a measurement range of 30 ° C to 57 ° C.*

***Keywords: Measurement of Temperature, Sensor LM35, Peltier Element, Android, Logger***

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT dan baginda besar rasulullah SAW atas segala rahmat dan anugerahnya sehingga penulis dapat menyelesaikan tugas akhir ini dengan judul :

### **“RANCANG BANGUN SISTEM PENGKONDISIAN SUHU MENGGUNAKAN ELEMEN PELTIER DENGAN SISTEM MONITORING PADA SMARTPHONE ANDROID”**

Tugas akhir ini disusun guna memenuhi persyaratan bagi seorang mahasiswa untuk memperoleh gelar Ahli Madya bidang studi instrumentasi, program studi D-3 Metrologi dan Teknik Instrumentasi, jurusan Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember

Penulis mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun secara tidak langsung dalam pengerjaan dan pembelajaran tugas akhir ini. Beberapa pihak tersebut antara lain:

1. Bapak Dr. Ir. Purwadi Agus Darwito, M.Sc dan Ir. Heri Joestiono, MT. selaku Dosen Pembimbing yang setia mendampingi, membimbing, mengkritisi, dan memotivasi pengerjaan tugas akhir ini.
2. Bapak Dr. Ir. Totok Soehartanto, DEA selaku Kepala Jurusan Teknik Fisika ITS Surabaya.
3. Bapak Dr. Ir. Purwadi Agus Darwito, M.Sc selaku Ketua Program Studi Diploma III Metrologi dan Teknik Instrumentasi, Institut Teknologi Sepuluh Nopember.
4. Bapak Ir. Sarwono, MM selaku Dosen Wali dan yang telah memberikan arahan selama menjalani masa perkuliahan hingga menyelesaikan tugas akhir ini.
5. Bapak dan Ibu dosen Jurusan Teknik Fisika beserta karyawan atas ilmu dan dedikasinya.

6. Teman teman D3 Metrologi dan Teknik Instrumentasi angkatan 2011, 2012, 2013, dan para senior yang selalu mendukung dalam segala hal.

Penulis menyadari bahwa tugas akhir ini tidaklah sempurna, tetapi penulis berharap ini dapat memberikan kontribusi yang berarti dan dapat menambah wawasan bagi pembaca. Semoga awal dari permulaan yang panjang ini dapat membawa manfaat dan hikmat bagi kita semua dan juga semoga hari esok lebih baik dari hari ini.

Surabaya, 17 Juli 2014

Penulis



## DAFTAR ISI

	Hal.
<b>LEMBAR JUDUL .....</b>	<b>i</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>iii</b>
<b>ABSTRAK .....</b>	<b>v</b>
<b>KATA PENGANTAR .....</b>	<b>vii</b>
<b>DAFTAR ISI .....</b>	<b>ix</b>
<b>DAFTAR GAMBAR .....</b>	<b>xi</b>
<b>DAFTAR TABEL .....</b>	<b>xiii</b>

### **BAB I PENDAHULUAN**

1.1	Latar Belakang.....	1
1.2	Permasalahan .....	2
1.3	Tujuan.....	2
1.4	Batasan Masalah .....	2
1.5	Sistematika Laporan .....	2

### **BAB II TINJAUAN PUSTAKA**

2.1	Konversi Energi Pada BTS Menggunakan Sistem Pendingin Arus Searah .....	5
2.2	Elemen <i>Peltier</i> .....	7
2.3	Sensor LM35 .....	10
2.4	Arduino Mega ADK .....	11
2.5	Relay .....	13
2.6	Metode Perhitungan Spesifikasi Alat .....	14
2.7	Sistem Android .....	16

### **BAB III PERANCANGAN DAN PEMBUATAN ALAT**

3.1	Gambaran Umum Tugas Akhir .....	21
3.2	Perancangan Perangkat Mekanik .....	21
3.3	Perancangan Perangkat Elektrik .....	22
	3.3.1 <i>Power Supply</i> .....	23
	3.3.2 Sensor Suhu (LM35) .....	24
	3.3.3 Elemen Pendingin .....	25
	3.3.4 Sistem Kontrol (Arduino Mega ADK).....	26
	3.3.5 Rangkaian <i>Display</i> 25 .....	28

3.3.6 Program Aplikasi Sistem.....	29
------------------------------------	----

#### **BAB IV ANALISA DATA DAN PEMBAHASAN**

4.1 Pengujian Sensor LM35 .....	31
4.1.1 Perhitungan Akurasi, Error dan Presisi .....	31
4.1.2 Perhitungan Standart Deviasi.....	35
4.1.3 Perhitungan Ketidakpastian .....	36
4.2 Pengujian pada Software Android .....	37
4.3 Pengujian Efek Peltier Terhadap Ruangn .....	39
4.4 Pembahasan .....	41

#### **BAB V KESIMPULAN DAN SARAN**

5.1 Kesimpulan .....	43
5.2 Saran .....	44

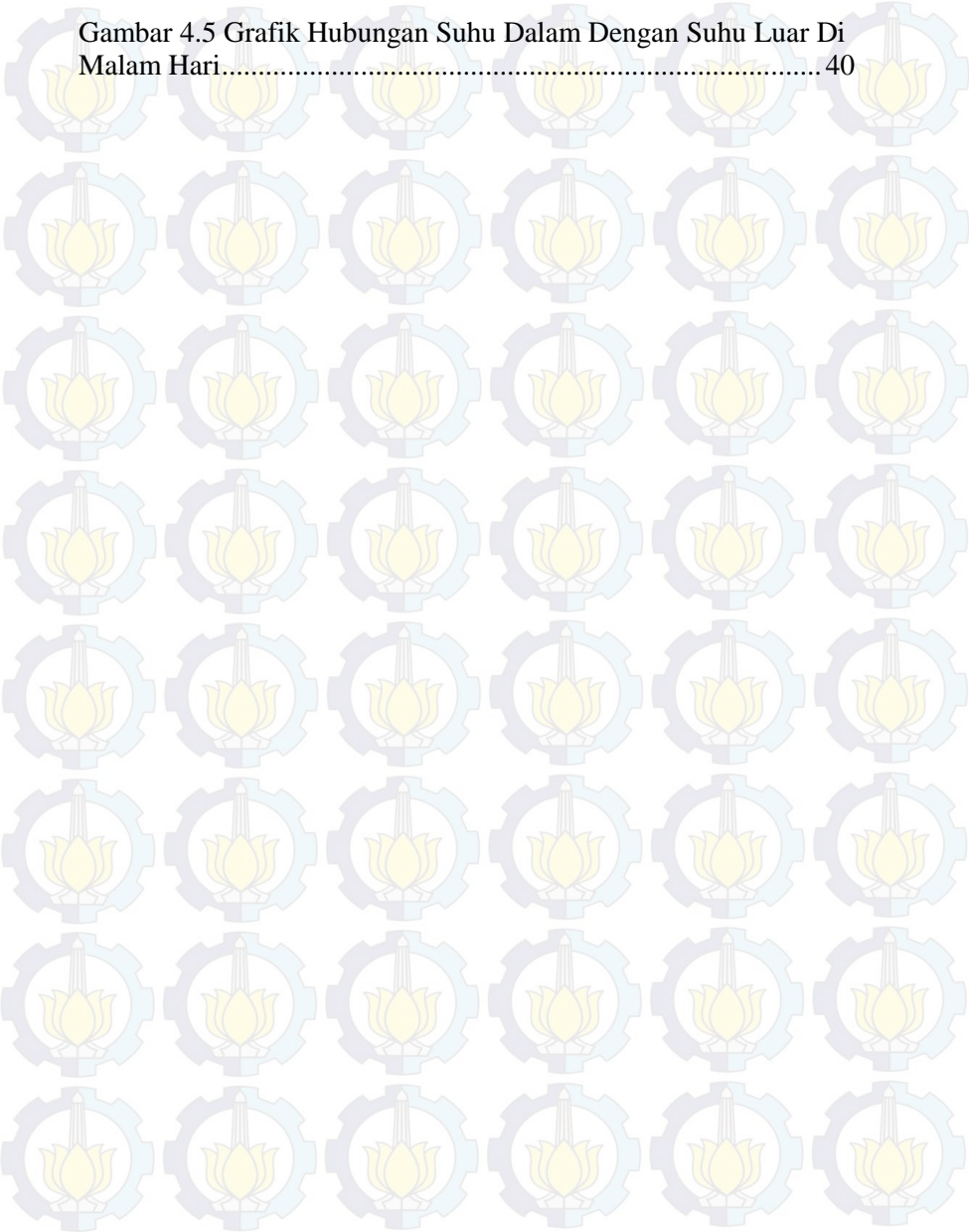
#### **DAFTAR PUSTAKA**

#### **LAMPIRAN**

## DAFTAR GAMBAR

	Hal.
Gambar 2.1 Susunan Dan Konstruksi DC Cooler .....	5
Gambar 2.2 DC Cooler Casing Pendek.....	6
Gambar 2.3 DC Cooler Casing Panjang.....	6
Gambar 2.4 DC Cooler Fan Besar.....	6
Gambar 2.5 DC Cooler Fan Kecil .....	7
Gambar 2.6 Pemasangan DC Cooler Pada Kabin .....	7
Gambar 2.7 Rangkaian Pengujian .....	7
Gambar 2.8 Elemen <i>Peltier</i> Sisi Panas.....	8
Gambar 2.9 Elemen <i>Peltier</i> Sisi Dingin.....	9
Gambar 2.10 Bagian-bagian Elemen <i>Peltier</i> .....	9
Gambar 2.11 Rangkaian <i>Heatsink</i> , Kipas, <i>Peltier</i> .....	10
Gambar 2.12 Sensor LM35 .....	11
Gambar 2.13 Arduino Mega ADK .....	12
Gambar 2.14 Skema Relay .....	13
Gambar 2.15 Logo Android .....	16
Gambar 2.16 komponen Sistem Android .....	17
Gambar 2.17 Sony X-peria E .....	18
Gambar 3.1 Flow Chart Metodologi Penelitian .....	19
Gambar 3.2 Diagram blok sistem pengkondisian suhu .....	21
Gambar 3.3 desain perangkat mekanik .....	22
Gambar 3.4 diagram fisik sistem.....	23
Gambar 3.5 rangkaian power supply.....	24
Gambar 3.6 Rangkaian Sensor LM35 .....	24
Gambar 3.7 Elemen <i>Peltier</i> .....	25
Gambar 3.8 Arduino Mega ADK .....	27
Gambar 3.9 Hasil Rangkaian.....	28
Gambar 3.10 Rangkaian Display.....	29
Gambar 4.1 Grafik Hubungan Suhu Dari Thermometer Dengan Suhu Dari LM35.....	34
Gambar 4.2 Tampilan Software Eclipse.....	37
Gambar 4.3 Data Hasil Pengukuran Suhu .....	38
Gambar 4.4 Grafik Hubungan Suhu Dalam Dengan Suhu Luar Di Siang Hari.....	40

Gambar 4.5 Grafik Hubungan Suhu Dalam Dengan Suhu Luar Di Malam Hari..... 40





## DAFTAR TABEL

Tabel 2.1 karakteristik Sensor LM35 .....	11
Tabel 3.1 Karakteristik Sensor LM35 .....	25
Tabel 3.2 Karakteristik Elemen <i>Peltier</i> .....	26
Tabel 3.3 Spesifikasi Arduino Mega ADK .....	26
Tabel 4.1 Hasil Pengujian LM35 .....	33
Tabel 4.2 Hasil Perhitungan Standart Deviasi .....	36
Tabel 4.3 Hasil Pengujian Plan <i>Smartroom</i> di Siang Hari .....	39
Tabel 4.4 Hasil Pengujian Plan <i>Smartroom</i> di Malam Hari .....	39

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Hampir di semua tempat dijumpai orang yang mengkonsumsi rokok. Berdasarkan analisa WHO (World Health organization), badan organisasi kesehatan dunia menunjukkan bahwa efek buruk asap rokok lebih besar bagi perokok pasif dibandingkan perokok aktif. Hasil kajian WHO<sup>[1]</sup>, lingkungan bebas asap rokok merupakan satu-satunya strategi efektif untuk memberikan perlindungan bagi perokok pasif.

Data Global Youth Survey tahun 1999-2006, sebanyak 81 persen anak usia 13-15 tahun di Indonesia menjadi perokok pasif. Survei tersebut juga menunjukkan, lebih dari 150 juta penduduk Indonesia menjadi perokok pasif di rumah, di perkantoran, di tempat umum dan juga di kendaraan umum.

Melihat keadaan tersebut, maka salah satu cara untuk mengurangi asap rokok agar tidak mengganggu orang lain yang tidak merokok adalah dengan memasang suatu alat untuk membantu membuang asap rokok. Salah satu cara yang dapat diambil adalah dengan menggunakan *fan exhaust* dan juga merancang sistem pendingin pada ruangan. Dalam perancangan sistem pendingin ini dapat digunakan elemen *peltier*.

Oleh karena itu, pada tugas akhir ini dirancang suatu alat yang memanfaatkan sensor LM35, elemen *peltier*, heatsink dan *fan exhaust*. Pada alat ini digunakan Arduino Mega ADK sebagai *controller* dan sistem monitoring suhu pada *smartphone* berbasis Android. Besar suhu yang terdapat pada ruangan dipengaruhi oleh besar suhu yang terdapat di luar ruangan. Prinsip kerja dari alat ini adalah ketika alat ini memperoleh tegangan maka elemen *peltier* dan sensor LM35 akan bekerja. Elemen *peltier* akan mendinginkan ruangan, sehingga terjadi perbedaan suhu antara suhu pada luar ruangan dengan suhu dalam ruangan. Sensor LM35 akan mengukur besar suhu yang terdapat di dalam ruangan. Sedangkan suhu yang terdapat di luar ruangan di ukur oleh termometer ruang, yang kemudian hasilnya dibandingkan.

Besar suhu yang dihasilkan oleh sensor LM35 akan ditampilkan pada *smartphone* berbasis Android.

### 1.2 Permasalahan

Permasalahan yang diangkat pada tugas akhir ini adalah melakukan eksperimen terhadap sistem pengkondisian udara di ruangan pada *prototype* yang berdimensi 2m x 1m x 2m dan suhu yang terdapat pada *prototype* tersebut dapat di-*monitoring* menggunakan *smartphone* berbasis Android.

### 1.3 Tujuan

Tujuan utama yang ingin dicapai dalam tugas akhir ini adalah untuk melakukan eksperimen terhadap alat pengkondisian udara di ruangan pada *prototype* yang berdimensi 2m x 1m x 2m dan suhu yang dihasilkan dapat di-*monitoring* menggunakan *smartphone* berbasis Android.

### 1.4 Batasan Masalah

Adapun batas ruang lingkup dari tugas akhir ini adalah sebagai berikut :

- Variabel yang diukur adalah suhu pada ruangan.
- Variabel yang dimanipulasi adalah suhu.
- *Prototype* ruangan yang digunakan berdimensi 2m x 1m x 2m.
- *Controller* yang digunakan adalah Arduino Mega ADK. Sensor suhu yang digunakan adalah LM35. Elemen pendingin yang digunakan adalah Elemen *Peltier*.
- Pengukuran suhu dilakukan setiap 3 menit sekali, baik ketika terdapat asap rokok maupun tidak terdapat asap rokok.
- Cara perancangannya adalah dengan menggunakan elemen *peltier*, sisi dingin dan sisi panas dari elemen ini dipasang *fan exhaust* dan heatsink yang berfungsi untuk mempercepat proses perpindahan, baik perpindahan panas maupun dingin.
- Hasil keluaran berupa nilai pengukuran, ditampilkan pada *smartphone* berbasis Android dalam bentuk *logger* yang berupa nilai hasil pengukuran suhu.



### **1.5 Sistematika Laporan**

Sistematika laporan yang digunakan dalam penyusunan laporan tugas akhir ini adalah sebagai berikut:

#### **BAB I PENDAHULUAN**

Berisi tentang latar belakang, permasalahan, tujuan, batasan masalah, dan sistematika laporan.

#### **BAB II TINJAUAN PUSTAKA**

Berisi tentang teori-teori tentang sistem pengkondisian udara pada ruangan berdimensi 2m x 1m x 2m.

#### **BAB III PERANCANGAN DAN PEMBUATAN ALAT**

Berisi tentang desain alat pengkondisian udara serta ruangan yang digunakan.

#### **BAB IV ANALISA DATA DAN PEMBAHASAN**

Berisi tentang analisa hasil pengujian alat pengkondisian udara dan suhu yang dihasilkan.

#### **BAB V PENUTUP**

Berisi tentang kesimpulan yang diperoleh dari analisa data dan saran.



**(Halaman ini sengaja dikosongkan)**

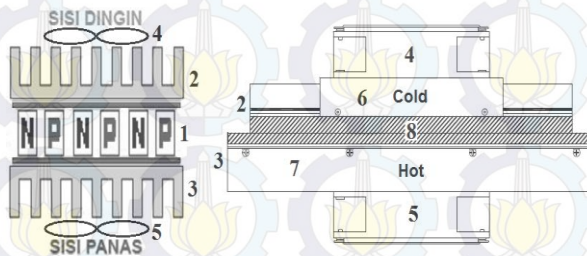
## BAB II TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan beberapa teori penunjang mengenai sistem pengkondisian udara pada ruangan berdimensi 2m x 1m x 2m, diantaranya adalah Elemen *Peltier*, Arduino Mega ADK, Sensor LM35, Relay, dan mengenai sistem Android.

### 2.1 Konversi Energi Pada BTS (*Base Transceiver Station*) Menggunakan Sistem Pendingin Arus Searah (*DC Cooler*)

Pada sistem telekomunikasi BTS berfungsi untuk menerima dan memancarkan sinyal komunikasi ke pelanggan. Pemakaian yang secara terus menerus akan menimbulkan panas pada sistem peralatan BTS (BTSE). Panas yang ditimbulkan dapat mempengaruhi kinerja alat dan umur alat. Oleh karena itu dibuat sistem pendingin agar mengurangi akumulasi panas serta mampu meningkatkan kinerja dan umur alat.<sup>[2]</sup>

Sistem pendingin yang digunakan memanfaatkan elemen *peltier*, dc fan dan juga *heatsink*. Pada eksperimen ini, *heatsink* dipasang pada sisi panas dan sisi dingin elemen *peltier*, begitu juga untuk dc fan yang digunakan.



**Gambar 2.1** Susunan dan konstruksi dc cooler

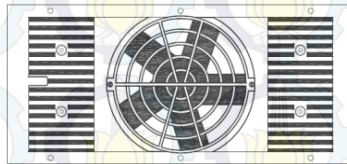
Berdasarkan gambar di atas, susunan dan konstruksi dc cooler terdiri dari:

1. Termoelektrik

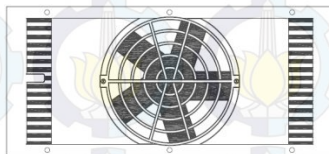
2. *Heatsink* dingin
3. *Heatsink* panas
4. Fan dingin
5. Fan panas
6. Casing dingin
7. Casing panas
8. Isolator

Eksperimen ini dilakukan pada kabin baterai berukuran 70 x 70 x 70 cm. Pengujian ini dilakukan terhadap beberapa variasi *prototype dc cooler* yang telah dirancang. Pengujian tersebut meliputi:

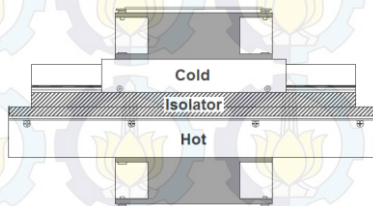
1. Casing pendek dan fan besar
2. Casing panjang dan fan kecil
3. Casing pendek dan fan kecil
4. Casing panjang dan fan besar



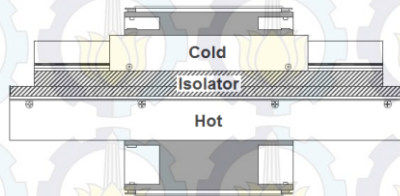
**Gambar 2.2** DC Cooler casing pendek



**Gambar 2.3** DC Cooler casing panjang

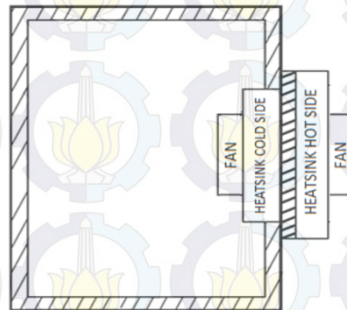


**Gambar 2.4** DC Cooler fan besar

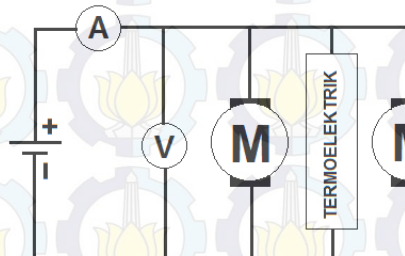


**Gambar 2.5** DC Cooler fan kecil

Dc cooler yang diuji dipasang pada kabin dan dirangkai seperti gambar berikut:



**Gambar 2.6** Pemasangan dc cooler pada kabin



**Gambar 2.7** Rangkaian pengujian

## 2.2 Elemen Peltier

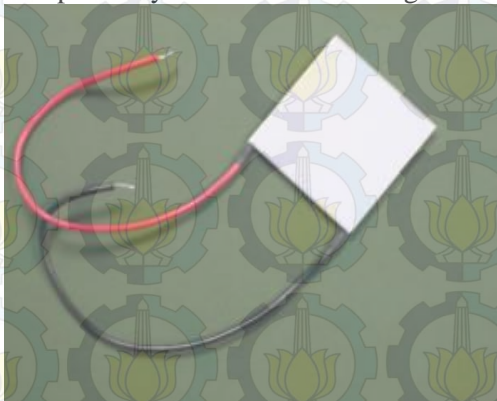
*Peltier* adalah suatu bahan yang terbuat dari semikonduktor, yang fungsinya untuk memindahkan panas dari prosesor ke elemen pendingin. Elemen pendingin yang dipakai dapat berupa



*heatsink* ataupun *waterblock* (jika menggunakan media air). Elemen *Peltier* merupakan modul/peralatan termoelektrik yang dapat mengubah energi listrik menjadi sebuah gradien temperatur. Sebagai sebuah sistem pendingin, elemen ini tidak bising, mudah perawatannya dan berdimensi relatif kecil, ringan serta ramah terhadap lingkungan karena tanpa refrigeran. Selain itu, *Peltier* adalah modul *Thermo-Electric*, umumnya dibungkus oleh keramik tipis yang berisikan batang-batang *Bismuth Telluride* di dalamnya.<sup>[3]</sup>

Fenomena termoelektrik pertama kali ditemukan tahun 1821 oleh ilmuwan Jerman, Thomas Johann Seebeck. Ia menghubungkan tembaga dan besi dalam sebuah rangkaian. Penemuan Seebeck ini memberikan inspirasi pada Jean Charles *Peltier* untuk melihat kebalikan dari fenomena tersebut, Maka hal ini dikenal dengan nama *Efek Peltier*. Ketika arus listrik dialirkan, terjadi penyerapan panas pada sambungan kedua logam tersebut dan pelepasan panas pada sambungan yang lainnya.<sup>[4]</sup>

Elemen *peltier* yang memiliki ukuran 40 x 40 x 3,8 mm ini terdiri dari 2 sisi yaitu, sisi dingin dan sisi panas. Pada umumnya, sisi yang panas dipasangkan dengan *heatsink* dan *fan exhaust* agar udara panas dapat terbuang. Namun, untuk mendinginkan ruangan pada sisi dingin dipasang pula *heatsink* dan *fan exhaust* agar udara dingin dapat menyebar ke seluruh ruangan.

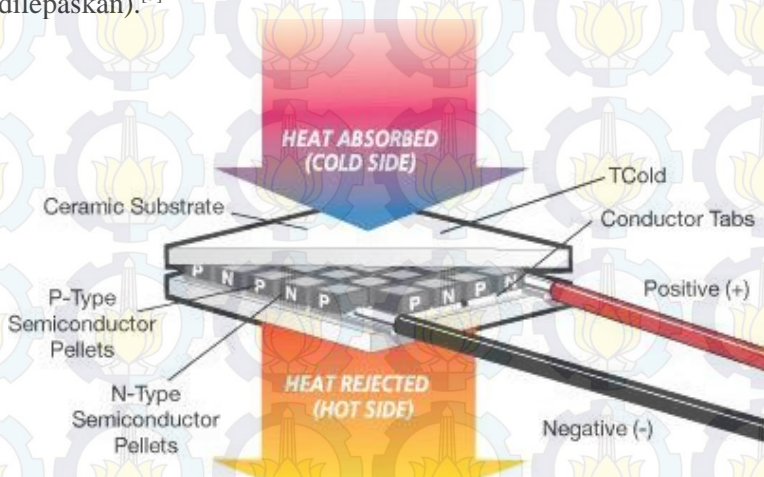


**Gambar 2.8** Elemen *Peltier* Sisi Panas



**Gambar 2.9** Elemen *Peltier* Sisi Dingin

Prinsip kerja dari elemen *peltier* yaitu ketika arus DC dialirkan ke elemen *peltier* yang terdiri dari beberapa pasang sel semikonduktor tipe *p* (semikonduktor yang mempunyai tingkat konduktivitas yang lebih rendah) dan tipe *n* (semi konduktor dengan tingkat konduktivitas yang lebih tinggi), akan mengakibatkan salah satu sisi elemen *peltier* menjadi dingin (kalor diserap) dan sisi lainnya menjadi panas (kalor dilepaskan).<sup>[5]</sup>



**Gambar 2.10** Bagian-bagian Elemen *Peltier*



**Gambar 2.11** Rangkaian *Heatsink*, Kipas dan *Peltier*

Pada gambar di atas terlihat bahwa kipas yang digunakan terdiri dari 4 buah dan diletakkan pada 2 (dua) buah sisi yaitu sisi panas dan sisi dingin. Kipas tersebut bertujuan membuang udara panas maupun dingin yang dihasilkan dari elemen *peltier*. Sehingga digunakan 4 *heatsink*, 4 *peltier* dan 4 kipas.

### 2.3 Sensor LM35

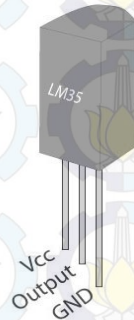
Sensor suhu LM35 adalah komponen elektronika yang memiliki fungsi untuk mengubah besaran suhu menjadi besaran listrik dalam bentuk tegangan. LM35 memiliki keakuratan tinggi dan kemudahan perancangan jika dibandingkan dengan sensor suhu yang lain, LM35 juga mempunyai keluaran impedansi yang rendah dan linieritas yang tinggi sehingga dapat dengan mudah dihubungkan dengan rangkaian kendali khusus serta tidak memerlukan penyetelan lanjutan.<sup>[6]</sup>

Prinsip kerja dari sensor LM35 adalah sebagai berikut:

- Suhu lingkungan di deteksi menggunakan bagian IC yang peka terhadap suhu
- Suhu lingkungan ini diubah menjadi tegangan listrik oleh rangkaian di dalam IC, dimana perubahan suhu berbanding lurus dengan perubahan tegangan output.



- Pada seri LM35 tiap perubahan  $1^{\circ}\text{C}$  akan menghasilkan perubahan tegangan output sebesar 10mV.



**Gambar 2.12** Sensor LM35

**Tabel 2.1** Karakteristik Sensor LM35

No.	Karakteristik	Nilai
1.	Sensitivitas	10mVolt/ $^{\circ}\text{C}$
2.	Akurasi Kalibrasi	$0.5^{\circ}\text{C}$
3.	Jangkauan Maksimal Operasi	$-55^{\circ}\text{C}$ sampai $+150^{\circ}\text{C}$
4.	Tegangan	4 – 30 Volt
5.	Arus Rendah	$< 60 \mu\text{A}$
6.	Ketidaklinieran	$\pm \frac{1}{4}^{\circ}\text{C}$

## 2.4 Arduino Mega ADK

Arduino adalah pengendali mikro single-board yang bersifat open-source, diturunkan dari *Wiring platform*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. Hardwarenya memiliki prosesor Atmel AVR dan softwarenya memiliki bahasa pemrograman sendiri.

Arduino mega Android Development Kit atau sering disebut Arduino Mega ADK merupakan salah satu jenis dari Arduino yang memiliki kelebihan dibandingkan dengan Arduino jenis lainnya dari segi *interface* terhadap perangkat *mobile*. Arduino Mega ADK memiliki soket tambahan berupa USB *host* yang



menyebabkan mikrokontroler ini cocok dijadikan *interface* dari *handphone* Android.



**Gambar 2.13** Arduino Mega ADK

Dengan menggunakan perangkat lunak Arduino kita dapat menuliskan program (disebut *sketches*), mengecek apakah terdapat kesalahan pemrograman hingga mengisikan program ke mikrokontroler pada papan Arduino. Pada perangkat lunak inilah terjadi proses *compiling*, yaitu konversi dari program yang kita tulis menjadi kode-kode yang dapat dimengerti oleh mikrokontroler. Perangkat lunak IDE Arduino ini dikembangkan berdasarkan perangkat lunak *open source processing* yang banyak digunakan untuk menghasilkan program perangkat lunak interaktif pada PC. Bahasa pemrograman Arduino yang berbasis bahasa C dan merupakan pengembangan dari bahasa pemrograman *Wiring*. Arduino dikembangkan oleh beberapa orang, yaitu , dengan tujuan utama yaitu untuk menyederhanakan pengembangan perangkat/produk interaktif dengan cara menyederhanakan bahasa pemrograman yang digunakan dan menyediakan kontroler yang dapat dengan mudah digunakan untuk banyak aplikasi umum, namun masih cukup memadai digunakan untuk menunjang aplikasi yang lebih kompleks.

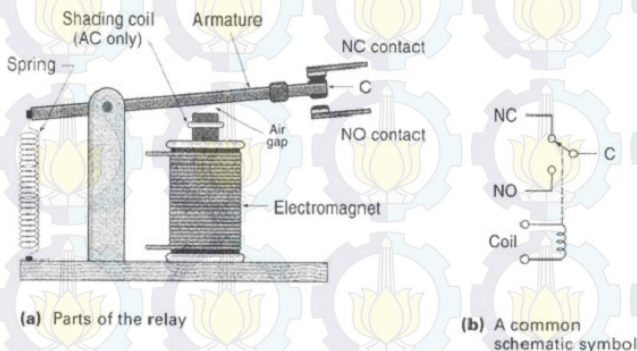
Arduino Mega ADK ini memiliki port USB yang dapat digunakan sebagai *connector* terhadap *handphone*, sehingga lebih fleksibel. Arduino jenis ini pada umumnya diperuntukkan untuk

sistem Android, sehingga integrasinya lebih mudah dan program yang digunakan tentunya lebih sedikit dan mudah dipahami oleh *user*. Perbedaan Arduino Mega ADK dengan Arduino pada umumnya hanyalah terletak pada banyaknya port digital yang digunakan. Arduino jenis Mega ADK ini memiliki *port digital* yang jauh lebih banyak dibandingkan dengan Arduino pada umumnya (Arduino Uno).

## 2.5 Relay

Relay adalah komponen yang terdiri dari sebuah kumparan berinti besi yang akan menghasilkan elektromagnet ketika kumparannya dialiri oleh arus listrik. Elektromagnet ini kemudian menarik mekanisme kontak yang akan menghubungkan kontak *Normally-Open* (NO) dan membuka kontak *Normally-Closed* (NC). *Normally-Open* (NO) adalah kontak yang pada saat Normal tidak terhubung, dan kontak *Normally-Closed* (NC) adalah kontak yang pada saat Normal terhubung.

Relay terdiri dari coil dan contact. Coil adalah gulungan kawat yang mendapat arus listrik, sedang contact adalah sejenis saklar yang pergerakannya tergantung dari ada tidaknya arus listrik di coil. Secara sederhana berikut ini prinsip kerja dari relay : ketika Coil mendapat energi listrik (*energized* ), akan timbul gaya elektromagnet yang akan menarik *armature* yang berpegas, dan contact akan menutup.<sup>[7]</sup>



**Gambar 2.14** Skema Relay Elektromekanik

Karakteristik relay antara lain adalah tegangan kerja koil/kumparan berkisar 5Vdc, 12Vdc, 24Vdc, 36Vdc, hingga 48Vdc. Tegangan kerja adalah tegangan yang harus diberikan kepada koil agar relay dapat bekerja. Selain itu ada karakteristik kemampuan kontak relay. Bisa 3A, 5A, 10A, atau lebih. Maksudnya disini adalah arus maksimal yang mampu dialirkan oleh kontak relay adalah sesuai dengan karakteristiknya, jadi bisa 3A, 5A, atau 10A.

Pada tugas akhir ini digunakan empat buah relay, relay tersebut berfungsi untuk mengontrol kerja dari *exhaust fan* dan elemen *peltier*.

## 2.6 Metode Perhitungan Spesifikasi Alat<sup>[8]</sup>

Untuk mengetahui spesifikasi suatu alat yang dibuat, hal pertama yang harus dilakukan yaitu menghitung standar deviasi. Standar deviasi merupakan ukuran penyebaran yang menunjukkan standar penyimpangan atau deviasi data terhadap rata-ratanya. Perhitungan standar deviasi ditunjukkan pada persamaan berikut :

$$STD = \sqrt{\frac{\sum (X_n - \bar{X}_n)^2}{n-1}} \quad (2.1)$$

Dimana :

STD = Standar Deviasi

$\bar{X}_n$  = Nilai hasil pengukuran

$X_n$  = Nilai rata-rata hasil pengukuran

N = Banyaknya data

Setelah standar deviasi didapatkan, langkah selanjutnya adalah menghitung ketidakpastian suatu alat. Ketidakpastian merupakan perkiraan atau taksiran rentang nilai pengukuran, dimana nilai sebenarnya dari obyek yang diukur terletak. Perhitungan ketidakpastian ditunjukkan pada persamaan berikut.

$$\text{Ketidakpastian} = \frac{STD}{\sqrt{n}} \quad (2.2)$$



Setelah nilai standar deviasi dan ketidakpastian didapatkan, langkah selanjutnya yaitu melakukan perhitungan besar *error* pengukuran. *Error* pengukuran merupakan penyimpangan nilai dari suatu pengukuran terhadap harga sebenarnya, dapat dinyatakan dalam prosen *error*. Perhitungan prosen *error* ditunjukkan pada persamaan berikut.

$$\text{Prosen } error \% = \left| \frac{Y_n - \bar{X}_n}{Y_n} \right| \times 100 \% \quad (2.3)$$

dimana :

$Y_n$  = Nilai sebenarnya

Setelah nilai standar deviasi, ketidakpastian, dan besar *error* diketahui, langkah selanjutnya adalah menghitung prosentasi nilai presisi. Nilai presisi merupakan keterdeteksian hasil pengukuran yang dilakukan berulang-ulang terhadap rata-rata pengukuran. Perhitungan prosen presisi ditunjukkan pada persamaan berikut.

$$\text{Prosen } presisi \% = 100\% - \left( \left| \frac{\bar{X}_n - \bar{X}_n}{\bar{X}_n} \right| \times 100 \% \right) \quad (2.4)$$

Setelah nilai standar deviasi, ketidakpastian, serta prosentasi *error* dan presisi didapatkan, langkah selanjutnya adalah menghitung nilai akurasi. Nilai akurasi merupakan keterdekatan hasil pengukuran terhadap nilai yang benar (*true value*) atau nilai standar yang disepakati. Perhitungan akurasi ditunjukkan pada persamaan berikut.

$$\text{Akurasi (A)} = 1 - \left| \frac{Y_n - \bar{X}_n}{Y_n} \right| \quad (2.5)$$

Dengan menghitung nilai standar deviasi, ketidakpastian, *error*, presisi, dan akurasi maka akan diketahui spesifikasi dari suatu alat yang dibuat dengan cara melihat hasil dari perhitungan-perhitungan karakteristik tersebut.



## 2.7 Sistem Android

Android merupakan system operasi pada perangkat mobile yang bersifat *open source* dan berbasis Linux. Karena Android bersifat *open source* dan mempunyai lisensi Apache yang sangat terbuka dan bebas, maka Android menjadi sistem operasi yang sangat populer bagi produsen perangkat mobile. Salah satu keunggulan Android adalah adanya komunitas para developer dan programmer yang luas untuk mengembangkan berbagai aplikasi yang berjalan di perangkat berbasis Android sehingga mampu memperluas fitur dan kemampuan perangkat tersebut.<sup>[9]</sup>

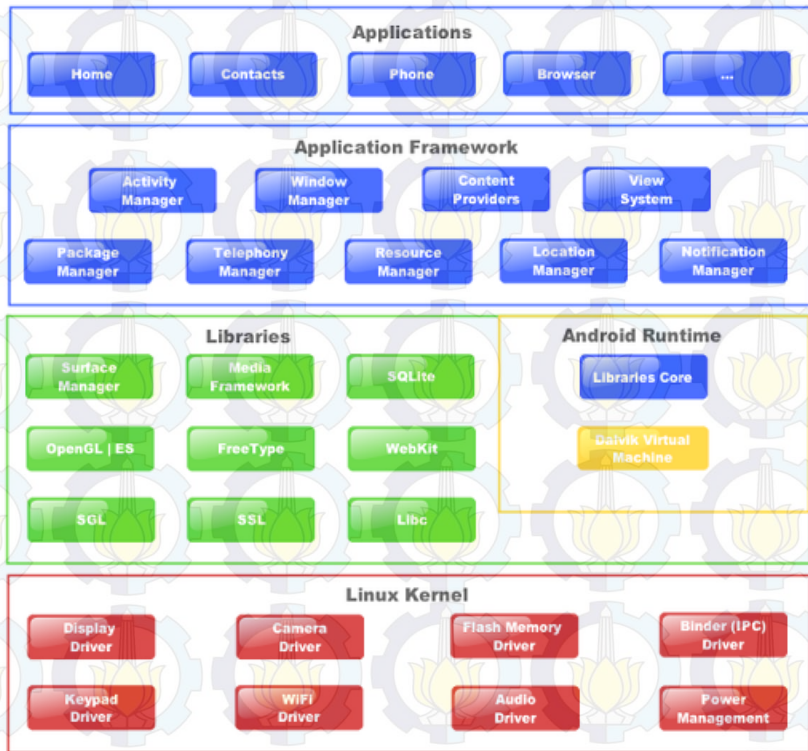


**Gambar 2.15** Logo Android

Android memiliki arsitektur yang terdiri dari:

1. *Application* dan *widget*
2. *Application Frameworks*
3. *Libraries*
4. *Android Run Time*
5. *Linux Kernel*

Komponen system Android bisa digambarkan dalam gambar berikut:



**Gambar 2.16** Komponen system Android

Software stack untuk Android OS terdiri atas berbagai aplikasi Java yang berjalan di atas *framework* Java berorientasi obyek, yang kemudian berjalan di atas *core library* Java dari Dalvik Virtual Machine (DVM) melalui *JIT compiler*. Sebagian *library* yang kritis ditulis dengan bahasa C demi kinerja seperti *Surface Manager* untuk dukungan *touchpad/touchscreen*, OpenCore Media Framework, SQLite untuk database, OpenGL ES untuk grafis 3D, FreeType untuk dukungan font, WebKit *layout engine* untuk *user interface*, SGL *graphics engine* untuk grafis 2D, SSL untuk koneksi yang aman, dan *Bionic libc*. Saat ini Android OS terdiri dari 12 juta baris kode termasuk 3 juta baris

kode XML, 2.8 juta baris kode dengan bahasa C, 2.1 juta baris kode dengan bahasa *Java*, dan 1.75 juta baris kode dengan bahasa C++. Pengembangan Android masih terus berlanjut dengan menambahkan berbagai fitur yang menarik, sehingga jumlah baris kode untuk Android akan terus bertambah dengan keluarnya versi-versi baru.

Android terdiri dari berbagai versi, diantaranya adalah versi 1.1, 1.5, 1.6, 2.2, 2.3, 3.0, 4.0, 4.1, dan masih banyak lainnya. Pada tugas akhir ini digunakan Android versi 4.1 (*Jelly Bean*). Android ini merupakan versi Android yang terbaru saat ini. Salah satu keunggulan dari Android versi ini adalah kemampuan *on-screen keyboard* lebih cepat dan lebih responsif.<sup>[10]</sup>

Pada tugas akhir ini digunakan telepon seluler dengan tipe Sony Xperia C-105 dengan Android versi 4.1 (*Jelly Bean*).



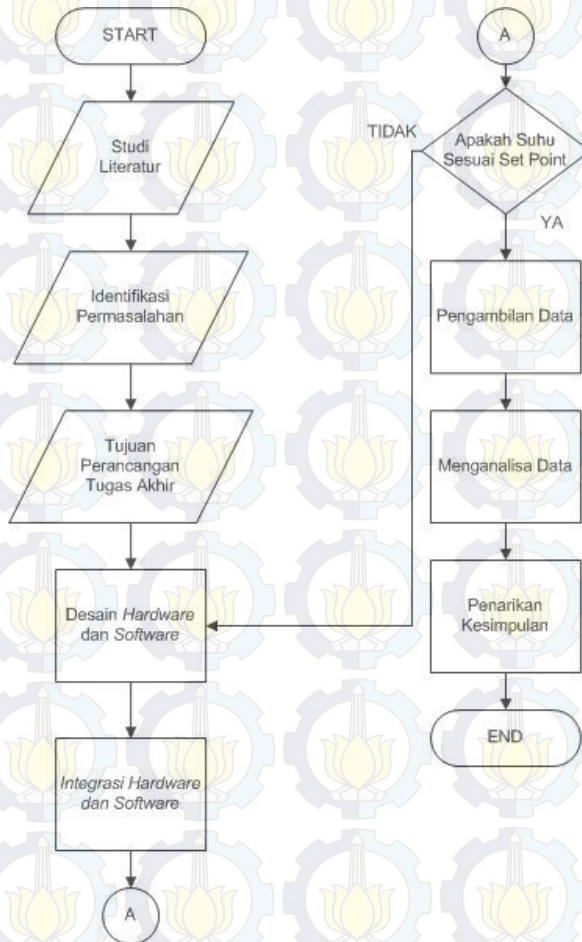
**Gambar 2.17** Sony Xperia C-105



### BAB III

## PERANCANGAN DAN PEMBUATAN ALAT

Dalam rancang bangun sistem pengkondisian udara di ruangan tertutup ini, metodologi yang digunakan dapat dijelaskan sesuai Gambar 3.1 berikut.

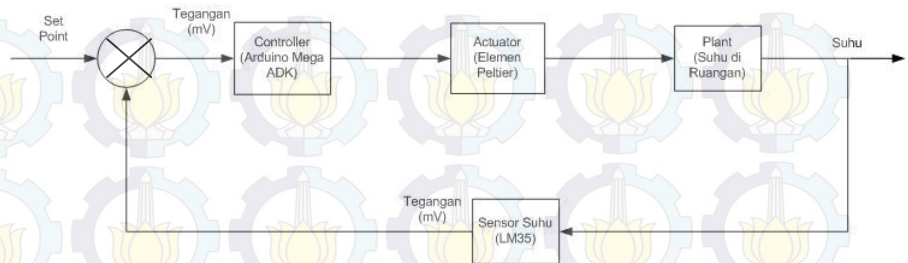


**Gambar 3.1** Flowchart dari Metodologi Penelitian



Dari Gambar 3.1 dapat dijelaskan *flowchart* metodologi penelitian adalah melalui tahap-tahap sebagai berikut. Tahapan pertama adalah melakukan *study literature* untuk merancang sistem yang dibutuhkan, dimana *study literature* yang dibutuhkan dalam penelitian ini adalah perkembangan sistem pengkondisian udara pada ruangan berasap rokok dan *me-monitoring* suhu yang dihasilkan dengan menggunakan *smartphone* berbasis Android. Tahap selanjutnya adalah mengidentifikasi permasalahan yang ada (sistem pengkondisian ruangan dan sistem Android). Pada saat ini banyak ruangan berasap rokok yang belum terdapat sistem pengkondisian di dalamnya. Sehingga hal tersebut dapat menyebabkan polusi udara pada ruangan karena adanya asap rokok. Dan dalam hal ini sistem Android digunakan untuk menampilkan hasil pengukuran suhu. Langkah selanjutnya adalah mengambil tujuan dari permasalahan yang ada dalam sistem pengkondisian ruangan dan sistem *monitoring* menggunakan Android ini, yaitu membuat sistem pengkondisi pada ruangan dan menampilkan hasil pengukuran suhu dengan menggunakan *smartphone* berbasis Android. Setelah itu barulah membuat desain sistem pengkondisian pada ruangan berdimensi 2m x 1m x 2m yang terdiri dari perancangan beserta desain *hardware* dan *software*. Setelah rancangan beserta desain *hardware* dan *software* selesai, langkah selanjutnya adalah mengintegrasikan *hardware* dan *software* sistem pengkondisian pada ruangan dengan dimensi yang telah ditentukan. Apabila sistem tersebut sudah sesuai, maka dapat dilakukan pengambilan data. Pengambilan data yang dilakukan adalah pengukuran suhu yang terdapat di dalam ruangan dan pengukuran suhu di luar ruangan. Data yang didapatkan selanjutnya akan dianalisa. Analisa data akan menjadi acuan untuk menarik kesimpulan terhadap kinerja sistem pengkondisian pada ruangan. Tahapan terakhir adalah penyusunan laporan dari hasil penelitian yang telah dibuat, yaitu sistem pengkondisian pada ruangan berdimensi 2m x 1m x 2m.

Berikut diagram blok pada rancang bangun pengkondisian udara di ruangan tertutup:



**Gambar 3.2** Diagram Blok Sistem Pengkondisian Udara di Ruang

Pada Gambar 3.2, digambarkan bahwa Sistem kerja pengendalian suhu di ruangan tertutup bertujuan untuk mengkondisikan suhu dalam ruangan. Pada pengendalian suhu ini digunakan sensor LM35 untuk mendeteksi suhu ruangan tersebut. Keluaran dari LM35 adalah berupa tegangan yang kemudian masuk ke ADC (*Analog Digital Converter*) pada Arduino Mega ADK, yang merupakan kontroler dari sistem pengendalian suhu. Selanjutnya Arduino Mega ADK memberikan perintah *on/off* ke aktuator melalui relay yang terhubung dengan elemen *peltier*. Elemen *peltier* ini dipengaruhi oleh suhu luar ruangan.

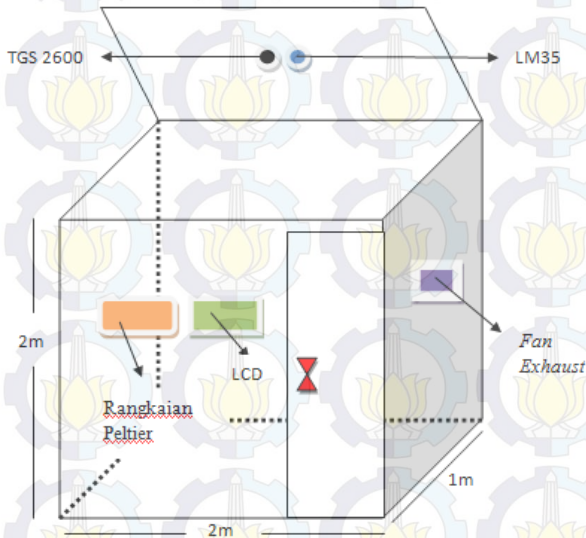
### 3.1 Gambaran Umum Tugas Akhir

Tugas akhir ini bertujuan untuk membuat sistem pengkondisian udara pada ruangan berdimensi 2m x 1m x 2m. Dimana pada ruangan tersebut terdapat pengukuran suhu baik ketika terdapat asap rokok maupun tidak terdapat asap rokok. Hasil pengukuran suhu yang diperoleh, ditampilkan pada *smartphone* berbasis Android.

### 3.2 Perancangan Perangkat Mekanik

Tahapan awal dalam perancangan tugas akhir kali ini adalah merancang desain mekanik dari sistem yang akan dibuat, sistem ini memanfaatkan desain ruangan pada umumnya yang dimodifikasi sesuai dengan sistem yang akan dibuat. Pada perancangannya, sensor LM35 akan diletakkan di bagian atas

(atap) dan rangkaian elemen *peltier* diletakkan di salah satu sisi dinding ruangan. Gambaran desain perangkat mekanik dapat dilihat pada Gambar 3.2 berikut.



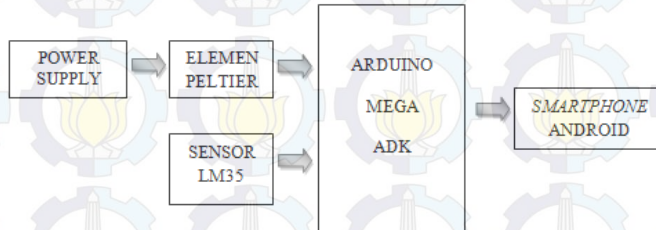
**Gambar 3.3** Desain perangkat mekanik dari pengkondisian pada ruangan

### 3.3 Perancangan Perangkat Elektrik

Pada rancangan sistem elektrik ini terdapat *power supply* yang berfungsi sebagai penyedia sumber tegangan agar semua komponen dalam sistem dapat bekerja. Sensor suhu yang digunakan adalah *LM35* dan elemen pendingin yang digunakan adalah elemen *peltier* yang dibantu dengan *fan exhaust*. Masing masing komponen tersebut (sensor dan elemen *peltier*) akan memberikan sinyal masukan berupa tegangan ke dalam rangkaian sistem Arduino Mega ADK untuk diolah dan hasil pengukuran suhu akan ditampilkan pada *smartphone* berbasis Android, dengan cara menggunakan konektor berupa kabel USB. Diagram fisik rancangan elektrik dari sistem pengkondisian ruangan



dengan sistem *monitoring* pada *smartphone* berbasis Android dapat dilihat pada gambar 3.4 berikut.



**Gambar 3.4** Diagram Fisik dari Sistem Pengkondisian Udara

Dari Gambar 3.3 dapat dijelaskan sebagai berikut :

- Input
  - *Sensor LM35*  
LM35 digunakan sebagai sensor suhu
  - *Elemen Peltier*  
Elemen *peltier* digunakan sebagai elemen pendingin
- Sistem Kontrol
  - *Arduino Mega ADK*  
Arduino Mega ADK sebagai pengolah data yang dikirimkan dari sensor LM35, dan sebagai *control* dari elemen *peltier*.
- Tampilan
  - *Smartphone berbasis Android*  
*Smartphone* berbasis Android digunakan untuk menampilkan data hasil pengukuran suhu dari sensor LM35 yang sudah diolah oleh Arduino.

### 3.3.1 Power Supply

Semua komponen dalam sistem ini tidak akan berfungsi jika tidak ada arus dan tegangan, oleh karena itu *power supply* (catu daya) digunakan sebagai penyedia sumber arus dan tegangan yang diperlukan oleh semua komponen yang ada dalam sistem





**Tabel 3.1.** Karakteristik Sensor LM35<sup>[12]</sup>

No.	Karakteristik	Nilai
1.	Sensitivitas	10mVolt/°C
2.	Akurasi Kalibrasi	0.5°C
3.	Jangkauan Maksimal Operasi	-55°C sampai +150°C
4.	Tegangan	4 – 30 Volt
5.	Arus Rendah	< 60 $\mu$ A
6.	Ketidaklinieran	$\pm \frac{1}{4}^{\circ}\text{C}$

### 3.3.3 Elemen Pendingin

Elemen pendingin yang digunakan pada sistem ini adalah elemen *peltier*. Elemen *peltier* ini memiliki dua buah sisi yang terdiri dari sisi panas dan sisi dingin. Elemen ini hanya mampu menerima tegangan maksimal sebesar 12 Volt DC.<sup>[13]</sup> Elemen ini dapat bekerja apabila dialiri arus maksimal 7A. Apabila terdapat aliran arus maka sisi panas *peltier* akan menjadi panas, begitu pula pada sisi dingin. Oleh karena itu, digunakan *heatsink* dan kipas (*fan exhaust*). *Heatsink* disini berfungsi untuk membuang suhu panas dan suhu dingin, sedangkan *fan exhaust* digunakan agar suhu panas dan dingin tersebut dapat menyebar. Gambar elemen *peltier* dapat dilihat pada gambar 3.7 berikut, sedangkan karakteristik sensor dapat ditunjukkan pada Tabel 3.2 di bawah.

**Gambar 3.7** Elemen *Peltier*

**Tabel 3.2.** Karakteristik Elemen *Peltier*

No.	Karakteristik	Nilai
1	Tegangan	12V.
2	Arus	4-7Amp.
3	Daya	72 Watt
4	Size	40 x 40 x 3.8 mm.

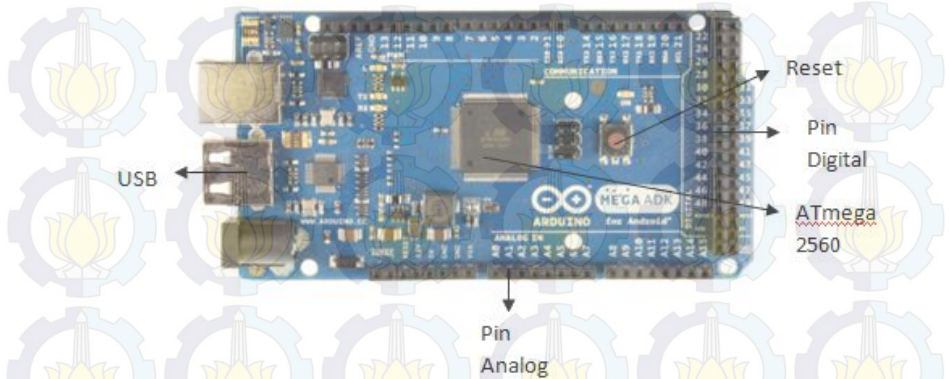
### 3.3.4 Sistem Kontrol ( Arduino Mega ADK )

Sistem kontrol pada sistem ini adalah menggunakan Arduino Mega ADK. Arduino ini bertugas untuk mengolah sinyal yang diberikan oleh sensor suhu untuk bisa ditampilkan pada *smartphone* berbasis Android. Di samping itu, juga sebagai sistem kontrol dari *fan exhaust* dan elemen *peltier*. Arduino yang digunakan adalah jenis Mega ADK dengan spesifikasi sebagai berikut.<sup>[14]</sup>

**Tabel 3.3** Spesifikasi Arduino Mega ADK

No.	Karakteristik	Nilai
1	Microcontroller	Atmega 2560
2	Operating Voltage	5 Volt
3	Input Voltage	7-12 Volt
4	Digital I/O Pin	54 pin
5	Analog Input Pin	16 pin
6	DC Current per I/O pin	40 mA
7	DC Current for 3.3 Volt Pin	50 mA
8	Flash Memory	256 Kb
9	SRAM	8 Kb
10	EEPROM	4 Kb
11	Clock Speed	16 MHz
12	USB Host Chip	MAX3421E





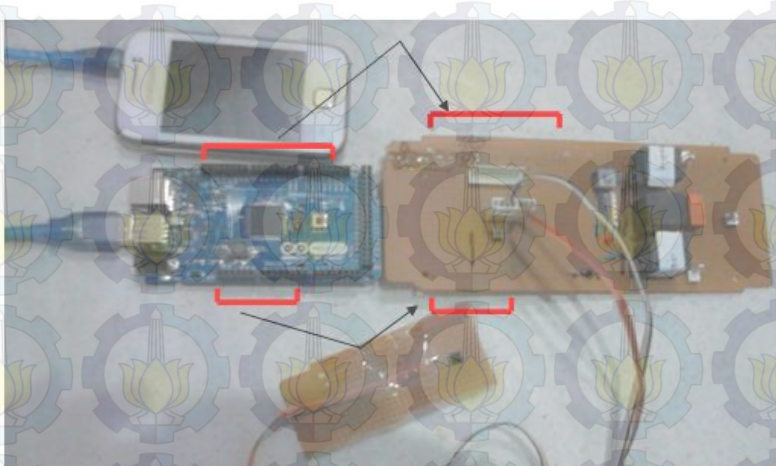
**Gambar 3.8** Arduino Mega ADK

Dari Gambar 3.8 dapat dijelaskan sebagai berikut.

- *Port input dan output Digital*, port yang digunakan untuk komunikasi data digital antara Arduino *board* dengan komponen lainnya baik secara masukan data seperti pengukuran dari sensor maupun mengirim data perintah ke komponen seperti menggerakkan motor servo.
- *Port input analog*, port yang digunakan untuk komunikasi analog antara sensor analog dengan Arduino *board*.
- *Port Universal Serial Bus (USB)*, port yang digunakan untuk komunikasi data antara Arduino *board* dengan komputer. Port ini dapat bersifat *input* maupun *output*, yaitu untuk meng-*upload* program ke memori Arduino, dan menerima data serial/data pengukuran sensor dari Arduino ke komputer.
- IC ATmega 2560, berfungsi untuk mengelola data dari sensor suhu.
- Tombol *reset*, berfungsi untuk mengembalikan Arduino ke kondisi awal.

Hasil rangkaian Arduino Mega ADK dapat dilihat pada Gambar 3.9 berikut.



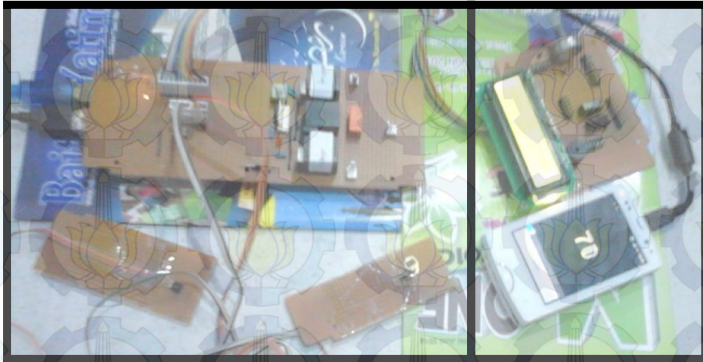


**Gambar 3.9** Hasil Rangkaian Arduino Mega ADK

Berdasarkan gambar 3.9, garis merah menunjukkan bahwa keduanya saling terhubung. Arduino Mega ADK dihubungkan dengan PCB yang telah dipasang rangkaian LM35 dan juga Elemen *Peltier*. Pin penghubung keduanya terletak di bagian bawah dari PCB.

### 3.3.5 Rangkaian Display ( *Smartphone* berbasis Android )

*Display* dalam sistem ini menggunakan *smartphone* berbasis Android. *Display* akan menerima sinyal yang telah diolah oleh Arduino Mega ADK sesuai dengan program yang telah dibuat. *Display* akan menampilkan variabel suhu ruangan sesuai dengan hasil pengukuran yang dilakukan. Rangkaian *display* pada *smartphone* berbasis Android dapat dilihat pada Gambar 3.10 di bawah ini.



**Gambar 3.10** Rangkaian *Display smartphone* berbasis Android

### 3.3.6 Program Aplikasi Sistem Pengkondisian Udara

Pemrograman sistem pengukuran suhu dan pengkondisian pada ruangan menggunakan Arduino Mega ADK. Prinsipnya, ketika alat ini memperoleh tegangan maka elemen *peltier* dan sensor LM35 akan bekerja. Elemen *peltier* akan mendinginkan ruangan, sehingga terjadi perbedaan suhu antara suhu pada luar ruangan dengan suhu dalam ruangan. Sensor LM35 akan mengukur besar suhu yang terdapat di dalam ruangan. Sedangkan suhu yang terdapat di luar ruangan di ukur oleh termometer ruang, yang kemudian hasilnya dibandingkan. Ketika sensor LM35 aktif, maka sensor akan mengirimkan sinyal ke Arduino Mega ADK untuk diolah dan hasil pengukuran ditampilkan pada *smartphone* berbasis Android. Pengukuran suhu dilakukan setiap 3 menit sekali. Sehingga pada *smartphone* akan ditampilkan hasil pengukuran suhu setiap 3 menit sekali.





## **BAB IV**

### **ANALISA DATA DAN PEMBAHASAN**

Pada bab ini dijelaskan mengenai pengujian alat pada ruangan tertutup berdimensi 2m x 1m x 2m. Setelah dilakukan perancangan dan pembuatan sistem pengendalian suhu pada ruangan tertutup maka perlu dilakukan pengujian terhadap *hardware* dan *software* yang telah dibuat. Selanjutnya dilakukan analisa secara menyeluruh agar dapat diketahui perfomansi alat secara keseluruhan yaitu dari *software* maupun *hardware*.

#### **4.1 Pengujian Sensor LM35**

Pengujian rangkaian sensor suhu LM35 bertujuan untuk mengetahui ketepatan sensor dalam merespon perubahan suhu. Pengujian tersebut dilakukan melalui proses perbandingan. Adapun cara pengujiannya yaitu :

1. Menentukan titik pengukuran yaitu 30, 33, 36, 39, 42, 45, 48, 51, 54, 57.
2. Melakukan pengukuran suhu dengan termometer digital dan mengukur tegangan keluaran (*Vout*) dari sensor LM35.
3. Mencatat hasil pengukuran
4. Mencari persamaan untuk sensor LM35 agar memiliki nilai yang relatif sama dengan nilai di temperatur.
5. Memasukkan nilai persamaan untuk sensor LM35 ke dalam program Arduino.
6. Menghitung koreksi, standart deviasi, akurasi, presisi dan *error*.
7. Menghitung ketidakpastian alat serta mengambil kesimpulan terhadap perfomansi alat tersebut.

##### **4.1.1 Perhitungan Akurasi, *Error* dan Presisi**

Akurasi merupakan selisih antara nilai pengukuran dengan nilai sebenarnya. Untuk mendapatkan nilai akurasi,



maka dapat dihitung dengan rumus berikut.

$$A = 1 - \left| \frac{Y_n - X_n}{Y_n} \right| \times 100\% \quad (4.1)$$

dengan :

A = Akurasi  
 $Y_n$  = Nilai Sebenarnya  
 $X_n$  = Nilai Pengukuran

*Error* merupakan penyimpangan nilai pengukuran terhadap nilai sebenarnya.

$$\text{Prosen error \%} = \left| \frac{Y_n - X_n}{Y_n} \right| \times 100 \% \quad (4.2)$$

Presisi merupakan keterdeteksian hasil pengukuran yang dilakukan berulang-ulang terhadap rata-rata pengukuran. Perhitungan prosen presisi ditunjukkan pada persamaan berikut.

$$\text{Prosen presisi \%} = 100\% - \left( \left| \frac{X_n - \bar{X}_n}{\bar{X}_n} \right| \times 100 \% \right) \quad (4.3)$$

Data hasil pengujian pada sensor LM35 dapat dilihat pada tabel 4.1 berikut.

Tabel 4.1. Hasil Pengujian LM35

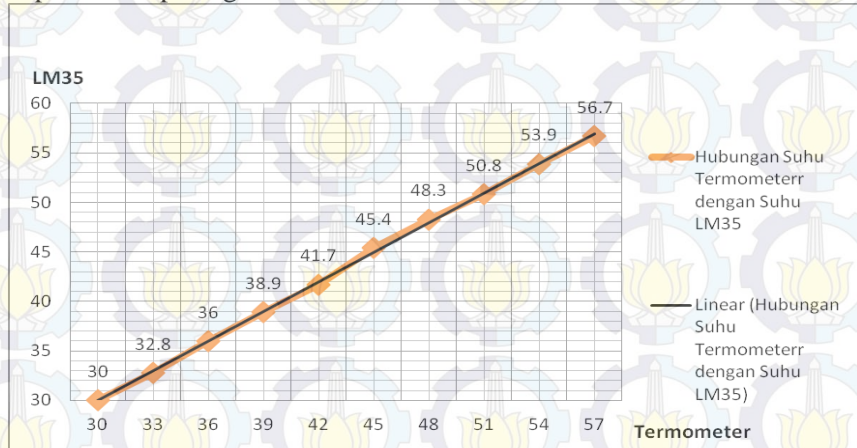
No.	Hasil Pengukuran			Akurasi (%)	Error	Presisi
	Suhu Termometer (°C)	Suhu LM35 (°C)	Vout LM35 (mV)			
1	30	30	301	100.00%	0	69.04%
2	33	32.8	336	99.39%	0.6	75.49%
3	36	36	361	100.00%	0	82.85%
4	39	38.9	376	99.74%	0.2	89.53%
5	42	41.7	417	99.28%	0.7	95.97%
6	45	45.4	448	99.11%	0.8	95.51%
7	48	48.3	482	99.37%	0.6	88.84%
8	51	50.8	511	99.61%	0.4	83.08%
9	54	53.9	535	99.81%	0.2	75.95%
10	57	56.7	573	99.47%	0.5	69.51%
Rata-rata				99.58%	0.04	82.58%

Berdasarkan tabel 4.1 di atas, diketahui bahwa pengukuran dilakukan pada titik 30, 33, 36, 39, 42, 45, 48, 51, 54, dan 57. Pada setiap titik pengukuran dilakukan pengulangan pengukuran sebanyak 5 kali, dan hasil *Vout* yang terdapat pada tabel 4.1 merupakan rata-rata dari hasil pengukuran di setiap titik pengukuran.

Di samping itu, melalui tabel 4.1 dapat diketahui bahwa rata-rata akurasi dari pengukuran adalah sebesar 99.58%, hal ini berarti bahwa alat tersebut memiliki nilai pengukuran yang cukup mendekati nilai pengukuran sebenarnya. Rata-rata *error* yang diperoleh adalah 0.04%, hal ini berarti bahwa hasil pengukuran tidak *linier*, karena terdapat penyimpangan. Sedangkan rata-rata presisi yaitu sebesar 82.58%, hal ini dipengaruhi oleh perbedaan rata-rata hasil pengukuran.

Grafik hubungan antara hasil pengukuran suhu dengan

sensor LM35 dan hasil pengukuran suhu dengan termometer dapat dilihat pada gambar 4.1 berikut.



**Gambar 4.1** Grafik Hubungan Suhu Termometer dengan Suhu LM35

Berdasarkan gambar 4.1 di atas, hasil grafik hubungan hasil pengukuran suhu menggunakan LM35 dengan termometer memiliki penyimpangan terhadap *linieritas*. *Linieritas* merupakan Hasil penyimpangan tersebut menunjukkan bahwa data hasil pengukuran kurang stabil, sehingga alat tersebut memiliki ketidaklinieran (*nonlinieritas*).

Di samping itu, berdasarkan data tersebut diperoleh persamaan regresi sebagai berikut.

$$y = 0.9978x + 0.0467 \quad (4.4)$$

dimana 0.9978 merupakan nilai gradient dari data tersebut. Hal ini membuktikan bahwa setiap kenaikan 1°C nilai tegangan naik sebesar 0.9978 mV (0.9978mV/°C). hasil ini hampir sesuai dengan karakteristik sensor LM35 yaitu setiap kenaikan suhu sebesar 1°C maka nilai tegangan akan naik sebesar 10.0 mV (10.0 mV/°C).



#### 4.1.2 Perhitungan Standart Deviasi

Standar deviasi merupakan ukuran penyebaran yang menunjukkan standar penyimpangan atau deviasi data terhadap rata-ratanya. Perhitungan standar deviasi ditunjukkan pada persamaan berikut :

$$STD = \sqrt{\frac{\sum (\bar{X}_n - \bar{X}_n)^2}{n-1}} \quad (4.5)$$

dengan :

STD = Standar Deviasi

$\bar{X}_n$  = Nilai hasil pengukuran

$\bar{X}_n$  = Nilai rata-rata hasil pengukuran

N = Banyaknya data

Hasil perhitungan standart deviasi sensor LM35 dapat dilihat pada tabel 4.2 berikut.



**Tabel 4.2** Hasil Perhitungan Standart Deviasi

No.	Suhu Termometer (°C)	Suhu LM35 (°C)	Koreksi (y)	$D = y - \bar{Y}$	$D^2$
1	30	30	0	-0.19	0.0361
2	33	32.8	0.2	0.01	0.0001
3	36	36	0	-0.19	0.0361
4	39	38.9	0.1	-0.09	0.0081
5	42	41.7	0.3	0.11	0.0121
6	45	45.4	0.4	0.21	0.0441
7	48	48.3	0.3	0.11	0.0121
8	51	50.8	0.2	0.01	0.0001
9	54	53.9	0.1	-0.09	0.0081
10	57	56.7	0.3	0.11	0.0121
<b>Rata-rata</b>			0.19	-	-
<b>Jumlah</b>			-	-	0.169
<b>Standart Deviasi</b>			-	-	0.137032

#### 4.1.3 Perhitungan Ketidakpastian

Ketidakpastian merupakan perkiraan atau taksiran rentang nilai pengukuran, dimana nilai sebenarnya dari obyek yang diukur terletak. Perhitungan ketidakpastian ditunjukkan pada persamaan berikut.

$$\text{Ketidakpastian} = \frac{STD}{\sqrt{n}} \quad (4.6)$$

Berdasarkan persamaan 4.5, maka hasil perhitungan ketidakpastian sensor LM35 adalah sebagai berikut.

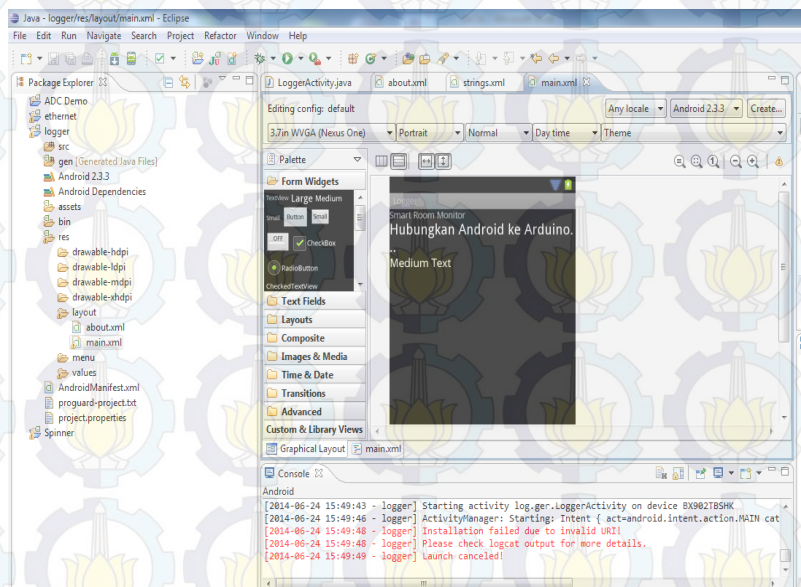
$$U = \frac{0.137032}{\sqrt{10}}$$

$$\frac{U}{U} = \frac{0.137032}{0.0433}$$

Berdasarkan perhitungan tersebut, maka nilai ketidakpastian dari sensor LM35 adalah 0.0433. Hal ini berarti bahwa alat tersebut memiliki nilai perkiraan atau rentang pengukuran sebesar 0.0433.

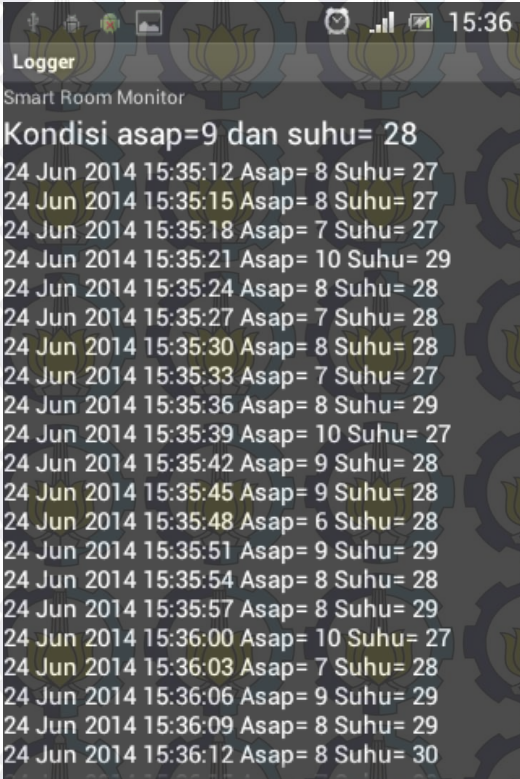
## 4.2 Pengujian pada *software* Android

Hasil pengukuran suhu ditampilkan dalam bentuk *logger* pada *smartphone* berbasis Android. Untuk bisa menampilkan hasil tersebut, dibuat *layout* program pada *software eclipse*. *Software* tersebut berfungsi untuk menghubungkan Arduino dengan Android, seperti pada gambar 4.1 berikut.



Gambar 4.2 Tampilan *Software Eclipse*

Sistem komunikasi antara Android dengan Arduino adalah menggunakan *USB connector*. Dengan begitu, Arduino dapat mengirimkan data melalui USB tersebut. Setelah Android dengan Arduino terhubung maka *logger* dapat digunakan pada *smartphone*. Data yang ditampilkan pada *logger* tersebut adalah setiap 3 menit sekali. Data tersebut adalah berupa nilai suhu dalam °C. hasil pengukuran suhu tersebut dapat dilihat pada gambar 4.2 berikut.



Kondisi asap=9 dan suhu= 28		
24 Jun 2014 15:35:12	Asap= 8	Suhu= 27
24 Jun 2014 15:35:15	Asap= 8	Suhu= 27
24 Jun 2014 15:35:18	Asap= 7	Suhu= 27
24 Jun 2014 15:35:21	Asap= 10	Suhu= 29
24 Jun 2014 15:35:24	Asap= 8	Suhu= 28
24 Jun 2014 15:35:27	Asap= 7	Suhu= 28
24 Jun 2014 15:35:30	Asap= 8	Suhu= 28
24 Jun 2014 15:35:33	Asap= 7	Suhu= 27
24 Jun 2014 15:35:36	Asap= 8	Suhu= 29
24 Jun 2014 15:35:39	Asap= 10	Suhu= 27
24 Jun 2014 15:35:42	Asap= 9	Suhu= 28
24 Jun 2014 15:35:45	Asap= 9	Suhu= 28
24 Jun 2014 15:35:48	Asap= 6	Suhu= 28
24 Jun 2014 15:35:51	Asap= 9	Suhu= 29
24 Jun 2014 15:35:54	Asap= 8	Suhu= 28
24 Jun 2014 15:35:57	Asap= 8	Suhu= 29
24 Jun 2014 15:36:00	Asap= 10	Suhu= 27
24 Jun 2014 15:36:03	Asap= 7	Suhu= 28
24 Jun 2014 15:36:06	Asap= 9	Suhu= 29
24 Jun 2014 15:36:09	Asap= 8	Suhu= 29
24 Jun 2014 15:36:12	Asap= 8	Suhu= 30

**Gambar 4.3** Data Hasil Pengukuran Suhu



### 4.3 Pengujian Efek Peltier Terhadap Ruangan

Pengujian *plan* pada proses ini bertujuan untuk mengetahui besar perubahan suhu yang terjadi ketika diberi efek dari elemen *peltier*. Proses pengujian ini dilakukan ketika siang hari dan malam hari, proses pengambilan data pada pengujian ini dilakukan 3 kali pengulangan pada setiap titik pengukuran, dengan *range* 25°C - 30°C. Pengukuran ini dilakukan dengan menggunakan termometer ruang. Adapun hasil data hasil pengujian dapat dilihat pada tabel 4.3 dan tabel 4.4 berikut.

**Tabel 4.3** hasil pengujian plan *smartroom* di siang hari

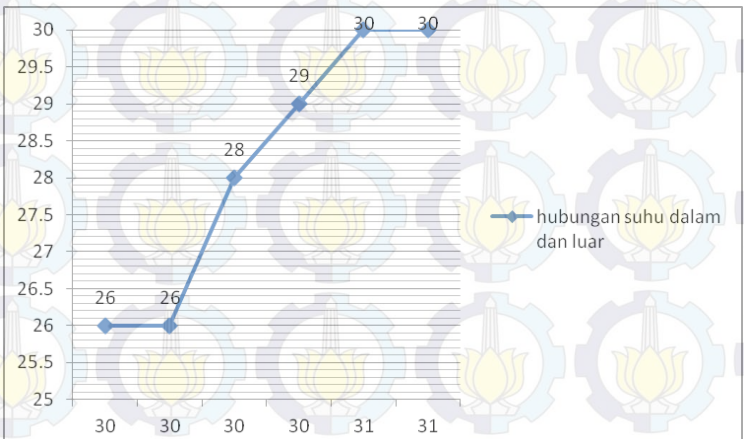
Siang			Error (%)
Set point	Suhu dalam	Suhu Luar	
25	26	30	0.13
26	26	30	0.13
27	28	30	0.06
28	29	30	0.03
29	30	31	0.03
30	30	31	0.03
Jumlah			0.41

**Tabel 4.3** hasil pengujian plan *smartroom* di malam hari

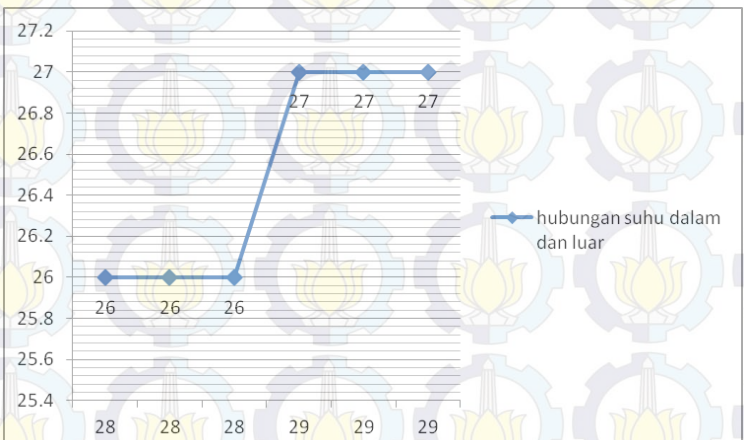
Malam			Error (%)
Set point	Suhu Dalam	Suhu Luar	
25	26	28	0.07
26	26	28	0.07
27	26	28	0.07
28	27	29	0.07
29	27	29	0.07
30	27	29	0.07
Jumlah			0.42



Berdasarkan tabel 4.3 dan tabel 4.4 di atas maka diperoleh grafik hubungan antara suhu dalam dengan suhu luar seperti gambar 4.4 dan gambar 4.5 berikut.



**Gambar 4.4** Grafik hubungan suhu dalam dengan suhu luar di siang hari



**Gambar 4.5** Grafik hubungan suhu dalam dengan suhu luar di malam hari

Berdasarkan gambar 4.4 dan gambar 4.5 di atas dapat diketahui bahwa besar suhu yang dihasilkan di ruangan tertutup dipengaruhi oleh suhu luar ruangan. Hal ini terlihat bahwa besar suhu yang dihasilkan rata-rata tidak sesuai dengan set point yang diberikan.

#### 4.4 Pembahasan

Laporan tugas akhir ini membahas tentang pengukuran suhu pada ruangan tertutup berdimensi 2mx1mx2m dan hasil pengukuran ditampilkan pada *smartphone* berbasis Android. Tujuan utama perancangan sistem ini adalah untuk mempermudah *monitoring* data berbasis *mobile*, untuk mempermudah dalam perancangan sistem pengkondisian udara di ruangan tertutup.

Perangkat keras yang digunakan antara lain sensor LM35 yang berfungsi untuk mengukur suhu, Arduino Mega ADK sebagai *controller*. Laptop berfungsi untuk memberikan tegangan pada Arduino. USB *connector* berfungsi sebagai sistem komunikasi dari Arduino ke Android. Sedangkan untuk perangkat lunak yang digunakan yaitu *Eclipse* yang berfungsi untuk merancang dan mengembangkan aplikasi Android. Pada *software* ini dibuat *layout* program serta pemberian *coding* untuk Android. Sehingga, *software* inilah yang digunakan dalam membuat *logger*.

Cara kerja dari sistem ini yaitu sensor akan mengukur suhu di ruangan tertutup kemudian outputnya ditransmisikan ke Arduino Mega ADK. Setelah itu, data diproses oleh *board* Arduino sehingga data analog keluaran dari sensor diubah menjadi digital. Data tersebut merupakan data hasil pengukuran yang kemudian ditampilkan pada *logger* di *smatrphone* berbasis Android.

Dari perhitungan perfomansi alat diperoleh data sebagai berikut. Pengambilan data ini dilakukan sebanyak 10 kali dengan *range* data 30°C - 57°C. Titik pada pengukuran tersebut yaitu 33, 36, 39, 42, 45, 48, 51, 54 dan 57. Pada setiap

pengukuran dilakukan pengulangan pengukuran sebanyak 5 kali. Kemudian, hasil pengukuran yang sudah tercatat dirata-rata. Data *Vout* hasil rata-rata itulah yang dimasukkan ke dalam data tersebut. Selanjutnya, Perhitungan akurasi sensor LM35 yang dihasilkan adalah sebesar 99.58%. Perhitungan *error* pada sensor LM35 adalah sebesar 0.04%. Perhitungan presisi sensor LM35 adalah sebesar 82.58%. Dari data tersebut dapat disimpulkan bahwa sensor LM35 memiliki *nonlinieritas* atau terjadi penyimpangan, sehingga data yang dihasilkan masih terdapat *error* yang cukup besar. Khususnya pada pengukuran suhu 45°C. Pada pengukuran tersebut terdapat *error* sebesar 0.8. *Error* ini terjadi karena alat atau sensor LM35 yang digunakan masih belum stabil atau konstan.

Kendala yang dialami dalam perancangan tugas akhir ini antara lain, *output* sensor LM35 tidak konstan, sehingga pengambilan data yang dilakukan cukup sulit, elemen pendingin yang digunakan masih belum dapat bekerja secara maksimal.



## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan percobaan dan analisa data, maka dapat ditarik kesimpulan pada tugas akhir ini sebagai berikut.

1. Telah berhasil dirancang alat pengkondisian udara pada ruangan tertutup berdimensi 2m x 1m x 2m dengan menggunakan sistem monitoring pada *smartphone* berbasis Android. Sensor Suhu yang digunakan adalah jenis sensor LM35, sedangkan untuk kontrolnya menggunakan Arduino Mega ADK dan layar tampilannya menggunakan *smartphone* berbasis Android.
2. Pengukuran suhu dilakukan setiap 3 menit sekali, dan hasil pengukuran ditampilkan pada *smartphone* berbasis Android dalam bentuk *logger*.
3. Dari hasil pengujian didapatkan rata-rata *error* sebesar 0.04% , rata-rata prosentasi presisi sebesar 82.58% dan rata-rata akurasi sebesar 99.58% dengan *range* pengukuran 30°C sampai 57°C.
4. Dari hasil pengujian efek elemen peltier terhadap ruangan tertutup dapat disimpulkan bahwa besar suhu yang dihasilkan di ruangan tertutup dipengaruhi oleh suhu luar ruangan tersebut.

#### **5.2 Saran**

Saran yang dapat disampaikan untuk ke depannya adalah:

1. Jumlah elemen peltier untuk ruangan yang lebih besar perlu diperbanyak.
2. Menggunakan termometer ruangan sebagai pembanding sensor LM35.



## DAFTAR PUSTAKA

- [1] Noorastuti, Pipit."Perokok Pasif".29 Januari 2014.  
[http://id.wikipedia.org/wiki/Perokok\\_pasif](http://id.wikipedia.org/wiki/Perokok_pasif)
- [2] Poetro, Joessianto Eko, M. Basuki Rahmat. (2012) .  
*Konservasi Energi Pada BTS Menggunakan Sistem Pendingin Arus Searah (DC Cooler)*. Surabaya: PPNS.
- [3] Putra, Nandy."Kotak Vaksin Untuk Daerah Pedalaman".  
29 Januari 2014.  
<http://nandyputra.blogspot.com/2006/12/penggunaan-heatsink-fan-sebagai.html>.
- [4] Sugiyanto. (2008). *Pengembangan Cool Box*. Depok:Universitas Indonesia.
- [5] Friansa, Koko. "Peltier". 29 Januari 2014.  
<http://www.slideshare.net/KokoFriansa/peltier-15352785>
- [6] Shatomeia. "Sensor Suhu LM35". 29 Januari 2014.  
<http://shatomeia.com/2008/12/sensor-suhu-lm35/>.
- [7] Wicaksono, Handy. (2009). *Relay –Prinsip dan Aplikasi*. Surabaya: Universitas Kristen Petra.
- [8] Hisyam, Ahmad. (2013). *Rancang Bangun Smart Inkubator : Alat Ukur Berat dan Panjang Bayi*. Surabaya: ITS
- [9] Pratama, Widiyanto. (2011). *Pengenalan Android*. Depok : Universitas Gunadharma.
- [10] Jamaludienz."Sistem Operasi Android". 25 Mei 2014.  
<http://jamaludienz.student.umm.ac.id/2011/09/30/sistem-operasi-android/>
- [11] Gunawan, Putu Nopa.(2011)."Power Supply".Makassar:Universitas Hasanuddin.
- [12] All Datasheet."Datasheet LM35". 29 januari 2014.<http://www.alldatasheet.com/view.jsp?Searchword=LM35>
- [13] Putra, Satria Sutardi. "Prinsip Peltier".29 Januari 2014.  
<http://satriasutardiputra.blogspot.com/>.
- [14] ADK, Arduino Board."Summary Arduino".30 Mei 2014.  
<http://arduino.cc/en/Main/ArduinoBoardMegaADK?from=Main.ArduinoBoardADK>



# LAMPIRAN A

## DATA PENGUJIAN SENSOR LM35 DAN PENGUJIAN EFEK PELTIER TERHADAP RUANGAN

### 1. Data Pengujian Sensor LM35

No.	Suhu Std	Pembacaan Vout LM35 berulang					Rata-rata Vout LM35
		1	2	3	4	5	
1	30	3.02	3.03	2.99	3.02	3.02	3.016
2	33	3.35	3.38	3.38	3.35	3.34	3.36
3	36	3.64	3.59	3.61	3.61	3.62	3.614
4	39	3.89	3.87	3.89	3.89	3.91	3.89
5	42	4.11	4.18	4.19	4.19	4.19	4.172
6	45	4.57	4.47	4.46	4.4	4.54	4.488
7	48	4.83	4.84	4.81	4.8	4.81	4.818
8	51	5.15	5.17	5.11	5.09	5.07	5.118
9	54	5.41	5.44	5.31	5.3	5.3	5.352
10	57	5.68	5.73	5.72	5.76	5.74	5.726

### 2. Data Pengujian Efek Peltier Terhadap Ruangan Di Siang Hari

Siang									
No.	Set Point	Suhu Dalam			Rata-rata	Suhu Luar			Rata-rata
		1	2	3		1	2	3	
1	25	26	26	26	26	30	30	30	30
2	26	26	26	26	26	30	30	30	30
3	27	27	28	28	27.66667	30	30	30	30
4	28	28	29	29	28.66667	30	30	30	30
5	29	30	30	30	30	31	31	31	31
6	30	30	30	31	30.33333	31	31	31	31

### 3. Data Pengujian Efek Peltier Terhadap Ruangan Di Malam Hari

Malam									
No.	Set Point	Suhu Dalam			Rata-rata	Suhu Luar			Rata-rata
		1	2	3		1	2	3	
1	25	26	25	26	25.66667	28	28	28	28
2	26	26	26	27	26.33333	28	28	28	28
3	27	27	26	26	26.33333	28	28	28	28
4	28	27	27	28	27.33333	29	29	29	29
5	29	27	27	27	27	29	29	29	29
6	30	28	27	27	27.33333	29	29	29	29

## LAMPIRAN B CODING PROGRAM

### 1. PROGRAM DI ARDUINO

```
// Library program
#include <LiquidCrystal.h>
#include <SPI.h>
#include <Adb.h>

// pin asap = A0
// pin suhu = A1

//variabel2
uint16_t suhu=0;
uint16_t asap=0;
uint16_t nol=1;
int ind=0;
float datasuhu[]={0,0,0,0};
float suhuf=0;
float asapf=0;
int ke=1;
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
Connection * connection;
long lastTime,lastTime2;
uint8_t LEDState=0;

//-----fungsi untuk menerima data dari android-----
void adbEventHandler(Connection * connection,
adb_eventType event, uint16_t length, uint8_t * data)
{
    if (event == ADB_CONNECTION_RECEIVE)
    {
        if(LEDState != data[0])
        {
            digitalWrite(21, data[0]);
            Serial.println(data[0],DEC);
        }
    }
}
```



LEDState = data[0];

```
}  
}  
}  
//-----  
//  
void setup()  
{  
  Serial.begin(9600);  
  
  lastTime = millis();  
  
  lcd.begin(16, 2);  
  pinMode(0,OUTPUT);  
  
  pinMode(14,OUTPUT); //  
  pinMode(15,OUTPUT); // BEL  
  pinMode(19,OUTPUT); // Saklar Power Peltier  
  pinMode(20,OUTPUT); // NC  
  pinMode(21,OUTPUT); // 24 VOLT  
  
  //inisialisasi koneksi ke android  
  ADB::init();  
  connection = ADB::addConnection("tcp:4568", true,  
  adbEventHandler);  
}  
  
void loop()  
{  
  if ((millis() - lastTime) > 500)  
  {  
    suhuf=analogRead(A1);  
    datasuhu[ind]=suhuf;  
    ind++; if(ind>3)ind=0;
```

```
suhuf=(datasuhu[0]+datasuhu[1]+datasuhu[2]+datasuhu[3])/4;
```

```
asapf=analogRead(A0);
```

```
Serial.print("Asap ");
```

```
Serial.print(asapf);
```

```
Serial.print(" = ");
```

```
//----- KALIBRASI ASAP -----
```

```
asapf=(asapf/1024)*0.583;
```

```
//-----
```

```
Serial.print(asapf);
```

```
Serial.print(" ");
```

```
Serial.print("suhu ");
```

```
Serial.print(suhuf);
```

```
Serial.print(" = ");
```

```
//----- KALIBRASI SUHU -----
```

```
suhuf=(5*(suhuf/1200)*100);
```

```
suhu=(uint16_t) suhuf;
```

```
//-----
```

```
Serial.println(suhu);
```

```
//---- LCD ----
```

```
lcd.clear();
```

```
lcd.setCursor(0, 0);
```

```
lcd.print("suhu: ");
```

```
lcd.print(suhu);
```

```
lcd.print(" C");
```

```
lcd.setCursor(0, 1);
```

```
lcd.print("asap: ");
```

```
lcd.print(asapf);
```

```

lcd.print("");
//-----
// ----- Keputusan ASAP-----

if(asap>100){
    digitalWrite(21,HIGH);
}
else{
    digitalWrite(21,LOW);
}
//-----
// ----- Keputusan SUHU-----
if(suhuf>32){
    digitalWrite(19,HIGH);
}
else{
    digitalWrite(19,LOW);
}
//-----
    lastTime = millis();
}

//-----bagian pengiriman paket data-----
if ((millis() - lastTime2) > 300)
{
    if(ke==1)connection->write(2,(uint8_t*)&nol);
    if(ke==2)connection->write(2,(uint8_t*)&asap);
    if(ke==3)connection->write(2,(uint8_t*)&suhu);
    ke++;
    if(ke>4)ke=1;
    lastTime2 = millis();
}
//-----
    ADB::poll();
}

```



## 2. PROGRAM DI ANDROID

```
package log.ger;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileOutputStream;
import java.io.FileReader;
import java.io.IOException;
import java.io.OutputStreamWriter;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import org.microbridge.server.AbstractServerListener;
import org.microbridge.server.Server;
import android.app.Activity;
import android.app.AlertDialog;
import android.graphics.Color;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class LoggerActivity extends Activity {
    /** Called when the activity is first created. */

    //-----INISIALISASI VARIABEL-----
    --
    private int adcSensorValue = 10, ke = 0;
    Server server = null;
```

```

int suhu;
int asap;
int sudah=0;
int minute,lastminute=0,pause=0;
String stat="Kondisi ";
TextView tv, scroll;
EditText ss;
//-----
private void baca() {
    File sdcard =
Environment.getExternalStorageDirectory();
    // Get the text file
    File file = new File(sdcard, "record.txt");
    // Read text from file
    StringBuilder text = new StringBuilder();
    //-----membaca per baris dimasukkan ke text-----
    try {
        BufferedReader br = new BufferedReader(new
FileReader(file));
        String line;

        while ((line = br.readLine()) != null) {
            text.append(line);
            text.append("\n");
        }
    } catch (IOException e) {
        // You'll need to add proper error handling here
    }

    // Set the text
    scroll.setText(text);
}

private void tulis(int stat) {
    try {
        File myFile = new File("/sdcard/record.txt");
        myFile.createNewFile();
        FileOutputStream fOut = new FileOutputStream(myFile);

```

```
OutputStreamWriter myOutWriter = new  
OutputStreamWriter(fOut);
```

```
SimpleDateFormat df = new SimpleDateFormat("d MMM yyyy  
HH:mm:ss");
```

```
String date = df.format(Calendar.getInstance().getTime());
```

```
if(stat==1){  
myOutWriter.append(scroll.getText()  
+ date + " Asap= " + asap + " Suhu= " + suhu);  
}
```

```
myOutWriter.close();
```

```
fOut.close();
```

```
baca();
```

```
} catch (Exception e) {
```

```
Toast.makeText(getBaseContext(), e.getMessage(),
```

```
Toast.LENGTH_SHORT)
```

```
.show();
```

```
}
```

```
}
```

```
//-----BLOK PROGRAM YANG DIEKSEKUSI  
PERTAMA-----
```

```
@Override
```

```
public void onCreate(Bundle savedInstanceState) {
```

```
super.onCreate(savedInstanceState);
```

```
setContentView(R.layout.main);
```

```
tv = (TextView) findViewById(R.id.textView1);
```

```
scroll = (TextView) findViewById(R.id.textView2);
```

```
baca();
```

```
try {
```

```
//Server adlh package untuk komunikasi dg arduino
```

```
server = new Server(4568);
```

```
server.start();
```

```
} catch (IOException e) {
```

```
Log.e("Seeeduino ADK", "Unable to start TCP server", e);
```

```
System.exit(-1);
```

```
}
```



```
server.addListener(new AbstractServerListener() {
    @Override
    public void onReceive(org.microbridge.server.Client client,
        byte[] data) {
        if (data.length < 2)
            return;
        adcSensorValue = (data[0] & 0xff) | ((data[1] & 0xff) << 8);

        runOnUiThread(new Runnable() {
            public void run() {
                //----- ketika nerima data suhu & asap
                new UpdateData().execute(adcSensorValue);
                //-----
            }
        });
    }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.menu, menu);
        return true;
    }
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.item1:
                Toast.makeText(getApplicationContext(), "Terhapus...", 200).show();
                return true;
            case R.id.item2:
                if (pause == 0) {
                    Toast.makeText(getApplicationContext(), "Log Paus", 200).show();
                    pause = 1;
                    stat = "(Pause) ";
                }
                else {
                    Toast.makeText(getApplicationContext(), "Log Start", 200).show();
                    pause = 0;
                    stat = "Kondisi ";
                }
            }
        }
    }
});
```

```

    }
    return true;
    case R.id.item3:
        showAbout();
        return true;
    default:
        return super.onOptionsItemSelected(item);
    }
}

class UpdateData extends AsyncTask<Integer, Integer, String> {
    @Override
    protected String doInBackground(Integer... sensorValue) {
        // SeekBar sbAdcValue = (SeekBar)
        findViewById(R.id.sbADCValue);
        // sbAdcValue.setProgress(sensorValue[0]);
        return (String.valueOf(sensorValue[0]))
    }

    @Override
    protected void onProgressUpdate(Integer... values) {
        super.onProgressUpdate(values);
    }

    @Override
    protected void onPostExecute(String result) {
        //data yang masuk ditampung ke variabel rcv //
        int rcv = Integer.parseInt(result);
        //bagian pemecahan / segmentasi data
        if (rcv == 1)
            ke = 0;
        if (ke == 1)
            asap = rcv;
        if (ke == 2)
            suhu = rcv;
        ke++;

        Calendar c = Calendar.getInstance();
        minute = c.get(Calendar.MINUTE);
    }
}

```

```
if (minute % 15 == 0 && sudah==0 && pause==0) {
    tulis(1);
    sudah=1;
    lastminute=minute;
}

if(minute!=lastminute){
    sudah=0;
}

tv.setText(stat + "asap=" + asap + " dan suhu=" + suhu);
}
    }
protected void showAbout
    // Inflate the about message contents
    View messageView = getLayoutInflater().inflate(R.layout.about,
    null, false);

    TextView textView2 = (TextView)
    messageView.findViewById(R.id.about_credits2);
    textView2.setTextColor(Color.WHITE);

    TextView textView = (TextView)
    messageView.findViewById(R.id.about_credits);
    textView.setTextColor(Color.WHITE);
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setIcon(R.drawable.ic_launcher);
        builder.setTitle(R.string.app_name);
        builder.setView(messageView);
        builder.create();
        builder.show();
    }
}
```



## BIODATA



Nama Penulis Andan Tunjung Pangesti, dilahirkan di Tulungagung, 28 Oktober 1993. Riwayat pendidikan penulis dimulai dari SDN 1 Ariyojeding dilanjutkan di SMPN 1 Rejotangan dan dilanjutkan lagi di SMAN 1 Rejotangan dan tahun 2011 masuk di prodi D3 Metrologi dan Instrumentasi Jurusan Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi

Sepuluh Nopember (ITS) Surabaya dengan NRP : 2411 031 015. Apabila terdapat pertanyaan tentang tugas akhir penulis maka dapat menghubungi No.Telpon Penulis yaitu : 085731972723, dan dapat juga melalui Email Penulis yaitu : andantunjung@gmail.com.